SPECIFICATION AMENDMENTS

Please replace the paragraph beginning at line 19 on page 5 with the following paragraph:

The algorithm first determines whether more than one key 16 is pressed in more than one column of keys, as shown by decision diamond ~~34~~ 38. For example, each time a key is pressed, a bit for that column is set. In one example, the right most bit in a four bit word is set if a key press value is registered for the first column. If the first column registers a set bit, then a "counter" is set to one, otherwise the counter remains at zero. The next column is then read to determine if it has a set bit or not. In the example above, the bit is set in the second position in the four bit word if a key press value is received for that column. The counter is advanced one if the bit is set, otherwise the counter remains at the previous value. The same process is performed in the next two columns of keys, where the set bit moves over to the left one position in the digital word each time. The counter is then read to determine if it is greater than one. If the counter is greater than one, then more than one key 16 has been pressed in more than one column, and an error is returned as shown by box 36. The algorithm then moves to a sub-routine that decodes dual key presses.

Please replace the paragraph beginning at line 1 on page 6 with the following paragraph:

In one non-limiting example, each key is assigned a key press value that is a hexa-decimal representation. If key #1 and key #2 are pressed at the same time, then the key press value is 0xffee. The NOT of 0xffee is ANDed with 0x000f, which is 1, so the counter is set to 1. The NOT of ~~the~~ 0xffee is shifted to the left one, and again ANDed with 0x000f, which is 1, so 1 is again added to the counter. The NOT of 0xffee is shifted to the left two and ANDed with 0x000f, which is 0, so 0 is added to the counter. The NOT of 0xffee is shifted to the left three bits and ANDed with 0x000f, which is 0, so 0 is added to the counter. Since this is the last column of keys, the counter is set at two, which is larger than 1, so the algorithm determines

that more than one key was pressed in more than one column. If only one of the two key #1 or key #2 had been pressed, then the counter would have a 1 at the end of the routine, indicating that only one key had been pressed in one column.

Please replace the paragraph beginning at line 23 on page 6 with the following paragraph(s):

Figure 3 is a decode table 50 including a plurality of cells 52 arranged in rows ~~54~~ 56 labeled A-D and columns ~~56~~ 54 labeled ~~0-3~~ 0-4, as shown. The cells 52 in rows A-D and columns 0-3 provide a one to one correspondence for each of the keys 16, where each cell 52 includes a particular number represented in hexa-decimal. The column 4 is a hexi-decimal start location for each row 56 used during adding of the key press values at box 46, discussed below.

Figure 4 is an exchange table 60 corresponding to the decode table 50. In one embodiment, the decode table 50 and the exchange table 60 are used to add the key press values on a row-by-row bases to determine a valid key value that will be transmitted to the terminal 22. It is stressed, however, that this is by way of a non-limiting example in that other techniques and digital representations of the key 16 can be employed within the teachings of the present invention.

Please replace the paragraph beginning at line 1 on page 7 with the following paragraph:

The algorithm then determines, at decision diamond ~~42~~ 40, if the accumulated key press value equals a predetermined valule, such as 1020, representing the added value for all of the keys pressed in all of the rows, including the added key press value for those keys that aren't pressed (which will be zero). If the answer is yes, the algorithm goes to box 36 to begin the dual key press sub-routine. If the answer is no, then the algorithm subtracts a predetermined value, such as 749, from the accumulated value, at box ~~42~~ 44, which is the key value that is transmitted to the terminal 22. Thus, the algorithm goes through the same number of steps regardless

if the first key is pressed or the last key is pressed to determine the key value. Therefore, a potential thief would not be able to determine the personal information of the user of the PIN pad 12 by recording the time it takes ~~to~~ the PIN pad 12 to determine which key 16 is pressed.